

IN THE CLAIMS:

The following listing of claims will replace all prior versions, and listings, of claims in the application.

1. - 42. (Cancelled)

43. (Currently Amended) A computer-accessible memory medium that stores program instructions executable by a processor to perform:

displaying a ~~data-acquisition (DAQ)~~ node in a graphical program;

receiving first user input invoking display of a plurality of ~~DAQ~~ functions for the ~~DAQ~~ node;

displaying the plurality of functions for the ~~DAQ~~ node in response to the first user input;

receiving second user input selecting a function from the plurality of ~~DAQ~~ functions;

determining graphical program code based on the second user input, wherein the determined graphical program code is executable to provide functionality in accordance with the selected function;

associating the determined graphical program code with the ~~DAQ~~ node, wherein, when the ~~DAQ~~ node in the graphical program executes, the determined graphical program code executes to provide the functionality in accordance with the selected function.

44. (Currently Amended) The memory medium of claim 43, wherein the ~~DAQ~~ node has a first node icon which is displayed in the graphical program, and wherein the first node icon has a first appearance, wherein the program instructions are further executable to perform:

changing the first node icon to a second appearance based on the second user input, wherein said changing the first node icon to a second appearance includes displaying an image corresponding to the selected function.

45. (Previously Presented) The memory medium of claim 43,
wherein said changing the first node icon to a second appearance comprises replacing the first node icon with a second node icon.

46. (Currently Amended) The memory medium of claim 43, wherein, prior to said associating the determined graphical program code with the ~~DAQ~~ node, the ~~DAQ~~ node does not have any associated graphical program code.

47. (Currently Amended) The memory medium of claim 43,
wherein, prior to said associating the determined graphical program code with the ~~DAQ~~ node, the ~~DAQ~~ node has associated default graphical program code in accordance with a default function for the node, and wherein the default graphical program code implement a first functionality; and

wherein said associating the determined graphical program code with the ~~DAQ~~ node comprises replacing the default graphical program code with the determined graphical program code.

48. (Currently Amended) The memory medium of claim 43,
wherein said receiving first user input comprises receiving the first user input to the ~~DAQ~~ node; and
wherein said receiving second user input comprises receiving the second user input to the ~~DAQ~~ node.

49. (Currently Amended) The memory medium of claim 43,
wherein said displaying the plurality of functions for the ~~DAQ~~ node in response to the first user input comprises;
displaying a plurality of function classes for the ~~DAQ~~ node; and
in response to user input selecting a function class, displaying the plurality of functions, wherein the plurality of functions are in the selected function class.

50. (Currently Amended) The memory medium of claim 43,
wherein the node is a data acquisition (DAQ) node;

wherein, prior to said associating, the DAQ node comprises one of:

- a generic read node;
- a generic write node;
- a generic channel creation node;
- a generic timing node; or
- a generic triggering node; and

wherein, after said associating, the DAQ node comprises one of:

- a specific read node in accordance with the selected function;
- a specific write node in accordance with the selected function;
- a specific channel creation node in accordance with the selected function;
- a specific timing node in accordance with the selected function; or
- a specific triggering node in accordance with the selected function.

51. (Currently Amended) The memory medium of claim 43, wherein the ~~DAQ~~ node represents a subprogram, wherein the program instructions are further executable to perform:

- receiving user input invoking expansion of the ~~DAQ~~ node; and
- displaying the subprogram in response to said invoking.

52. (Currently Amended) A computer-implemented method for configuring a graphical program node, comprising:

- displaying a ~~data acquisition (DAQ)~~ node in a graphical program;
- receiving first user input invoking display of a plurality of ~~DAQ~~ functions for the ~~DAQ~~ node;
- displaying the plurality of functions for the ~~DAQ~~ node in response to the first user input;
- receiving second user input selecting a function from the plurality of ~~DAQ~~ functions;

determining graphical program code based on the second user input, wherein the determined graphical program code is executable to provide functionality in accordance with the selected function;

associating the determined graphical program code with the ~~DAQ~~ node, wherein, when the ~~DAQ~~ node in the graphical program executes, the determined graphical program code executes to provide the functionality in accordance with the selected function.

53. (Currently Amended) A computer-accessible memory medium that stores program instructions executable by a processor to perform:

displaying a ~~data-acquisition (DAQ)~~ node in a graphical program;

receiving first user input invoking display of a plurality of ~~DAQ~~ functions for the ~~DAQ~~ node;

displaying the plurality of functions for the ~~DAQ~~ node in response to the first user input;

receiving second user input selecting a function from the plurality of ~~DAQ~~ functions;

determining a second node based on the selected function, wherein the second node comprises graphical program code executable to provide functionality in accordance with the selected function;

replacing the ~~DAQ~~ node in the graphical program with the second node, wherein, when the second node in the graphical program executes, the determined graphical program code executes to provide the functionality in accordance with the selected function.

54. (Currently Amended) The memory medium of claim 53, wherein the ~~DAQ~~ node comprises a first node icon, and wherein said displaying the ~~DAQ~~ node comprises displaying the first node icon, and wherein the second node comprises:

the first node icon and the determined graphical program code; or

a second node icon and the determined graphical program code.

55. (Currently Amended) The memory medium of claim 53, wherein the ~~DAQ~~ node and/or the second node is one or more of:

- polymorphic;
- function switchable; or
- function class switchable.

56. (Currently Amended) The memory medium of claim 53,
wherein the node is a data acquisition (DAQ) node;

wherein the DAQ node comprises one of:

- a generic read node;
- a generic write node;
- a generic channel creation node;
- a generic timing node; or
- a generic triggering node; and

wherein the second node comprises a corresponding one of:

- a specific read node in accordance with the selected function;
- a specific write node in accordance with the selected function;
- a specific channel creation node in accordance with the selected function;
- a specific timing node in accordance with the selected function; or
- a specific triggering node in accordance with the selected function.

57. (Previously Presented) The memory medium of claim 53, wherein the second node represents a subprogram, wherein the program instructions are further executable to perform:

- receiving user input invoking expansion of the second node; and
- displaying the subprogram in response to said invoking.

58. (Currently Amended) A computer-implemented method for configuring a graphical program node, comprising:

displaying a ~~data acquisition (DAQ)~~ node in a graphical program;
receiving first user input invoking display of a plurality of ~~DAQ~~ functions for the ~~DAQ~~ node;
displaying the plurality of functions for the ~~DAQ~~ node in response to the first user input;
receiving second user input selecting a function from the plurality of ~~DAQ~~ functions;
determining a second node based on the selected function, wherein the second node comprises graphical program code executable to provide functionality in accordance with the selected function;
replacing the ~~DAQ~~ node in the graphical program with the second node, wherein, when the second node in the graphical program executes, the determined graphical program code executes to provide the functionality in accordance with the selected function.

59. (Currently Amended) A computer-accessible memory medium that stores program instructions executable by a processor to perform:

displaying a ~~DAQ~~ node in a graphical program, wherein the ~~DAQ~~ node has a first node icon which is displayed in the graphical program, and wherein the first node icon has a first appearance;
receiving user input specifying one or more inputs to the ~~DAQ~~ node;
determining a function from a plurality of possible ~~DAQ~~ functions for the ~~DAQ~~ node based on the specified one or more inputs;
determining graphical program code based on the determined function, wherein the determined graphical program code is executable to provide functionality in accordance with the determined function;
associating the determined graphical program code with the ~~DAQ~~ node, wherein, when the ~~DAQ~~ node executes in the graphical program, the determined graphical program code is operable to execute to provide the functionality in accordance with the determined function.

60. (Currently Amended) The memory medium of claim 59, wherein the ~~DAQ~~ node has a first node icon which is displayed in the graphical program, and wherein the first node icon has a first appearance, wherein the program instructions are further executable to perform:

changing the first node icon to a second appearance based on the second user input, wherein said changing the first node icon to a second appearance includes displaying an image corresponding to the selected function.

61. (Previously Presented) The memory medium of claim 59,

wherein said changing the first node icon to a second appearance comprises replacing the first node icon with a second node icon.

62. (Currently Amended) The memory medium of claim 59, wherein, prior to said associating the determined graphical program code with the ~~DAQ~~ node, the ~~DAQ~~ node does not have any associated graphical program code.

63. (Currently Amended) The memory medium of claim 59,

wherein, prior to said associating the determined graphical program code with the ~~DAQ~~ node, the ~~DAQ~~ node has associated default graphical program code in accordance with a default function for the node, and wherein the default graphical program code implement a first functionality; and

wherein said associating the determined graphical program code with the ~~DAQ~~ node comprises replacing the default graphical program code with the determined graphical program code.

64. (Currently Amended) The memory medium of claim 59,

wherein the node is a data acquisition (DAQ) node;

wherein, prior to said associating, the DAQ node comprises one of:

a generic read node;

a generic write node;

- a generic channel creation node;
- a generic timing node; or
- a generic triggering node; and

wherein, after said associating, the DAQ node comprises one of:

- a specific read node in accordance with the selected function;
- a specific write node in accordance with the selected function;
- a specific channel creation node in accordance with the selected function;
- a specific timing node in accordance with the selected function; or
- a specific triggering node in accordance with the selected function.

65. (Currently Amended) The memory medium of claim 59, wherein the ~~DAQ~~ node represents a subprogram, wherein the program instructions are further executable to perform:

- receiving user input invoking expansion of the ~~DAQ~~ node; and
- displaying the subprogram in response to said invoking.

66. (Currently Amended) A computer-implemented method for configuring a graphical program node, comprising:

- displaying a ~~DAQ~~ node in a graphical program, wherein the ~~DAQ~~ node has a first node icon which is displayed in the graphical program, and wherein the first node icon has a first appearance;

- receiving user input specifying one or more inputs to the ~~DAQ~~ node;

- determining a function from a plurality of possible ~~DAQ~~ functions for the ~~DAQ~~ node based on the specified one or more inputs;

- determining graphical program code based on the determined function, wherein the determined graphical program code is executable to provide functionality in accordance with the determined function;

- associating the determined graphical program code with the ~~DAQ~~ node, wherein, when the ~~DAQ~~ node executes in the graphical program, the determined graphical program code is operable to execute to provide the functionality in accordance with the determined function.

67. (Currently Amended) A computer-accessible memory medium that stores program instructions executable by a processor to perform:

- displaying a ~~DAQ~~ node in a graphical program, wherein the ~~DAQ~~ node has a first node icon which is displayed in the graphical program, and wherein the first node icon has a first appearance;

- receiving user input specifying one or more inputs to the ~~DAQ~~ node;

- determining a function from a plurality of possible ~~DAQ~~ functions for the ~~DAQ~~ node based on the specified one or more inputs;

- determining a second node based on the selected function, wherein the second node comprises graphical program code executable to provide functionality in accordance with the selected function;

- replacing the ~~DAQ~~ node in the graphical program with the second node, wherein, when the second node in the graphical program executes, the determined graphical program code executes to provide the functionality in accordance with the selected function.

68. (Currently Amended) A computer-implemented method for configuring a graphical program node, comprising:

- displaying a ~~DAQ~~ node in a graphical program, wherein the ~~DAQ~~ node has a first node icon which is displayed in the graphical program, and wherein the first node icon has a first appearance;

- receiving user input specifying one or more inputs to the ~~DAQ~~ node;

- determining a function from a plurality of possible ~~DAQ~~ functions for the ~~DAQ~~ node based on the specified one or more inputs;

- determining a second node based on the selected function, wherein the second node comprises graphical program code executable to provide functionality in accordance with the selected function;

replacing the ~~DAQ~~ node in the graphical program with the second node, wherein, when the second node in the graphical program executes, the determined graphical program code executes to provide the functionality in accordance with the selected function.